Remarks/Arguments:

The above Amendments and these Remarks are in reply to the Office Action mailed May 20, 2005.

Claims 1 – 20 were pending in the Application prior to the outstanding Office Action. In the Office Action, the Examiner rejected claims 1-20. The present Response amends claims 1, 4, 11, 13 – 14, 18 and 20 to make explicit what was previously implicit and to correct minor informalities, leaving for the Examiner's present consideration claims 1 - 20. No new matter is added. Reconsideration of the rejections is requested.

Rejections under 35 U.S.C. § 102(b)

Claim 1

In items 3-11 on pages 2 - 5, the Office Action rejected claims 1, 4 – 11 and 13 – 15 and 20 as being anticipated by Nageswaran (US 5,991,792). The rejections are respectfully traversed.

Apparently, the Office Action's conclusion is based upon the assumption that applicant's claimed embodiments are limited to managing re-usable connections only, like Nageswaran, and that therefore, Nageswaran's "idle threads" could be read as applicant's recited "unavailable threads". (See, e.g., Office Action, page 9: "[and the thread manager 132 would identify these idle threads (i.e., read as unavailable threads) in the idle thread queue 140 by their thread ID, and mark their state as "Being Removed" (i.e., reducing resources in an unavailable queue)]"). If such a proposition were taken as arguendo true, then the Office Action's conclusions would indeed be plausible.

However, Nageswaran's approach cannot possibly provide the capability to manage a pool of connections in which resources that have been determined to be either not created successfully or not able to be refreshed are dequed separately in an unavailable deque from resources that have been determined to be available in an available deque as provided for by applicant's recited embodiments. (See, e.g., Specification, paragraph [0019]: "RESERVED contains resource instances that have been requested or are in use, AVAILABLE contains

resource instances currently available in the resource pool. UNAVAILABLE will contain resource instances that were previously not created successfully or able to be refreshed....").

Accordingly, to eliminate such confusion, claim 1 has been amended to make clear such capability:

> 1. (Currently Amended) A method for performing resource pool size maintenance for an application server, comprising:
>
> triggering a resource pool shrink check;
>
> determining that pool shrinking is necessary;
>
> reducing resources that have been determined to be at least one of not created successfully and not able to be refreshed in an unavailable deque; and
>
> reducing resources that have been determined to be available in an available deque.

Not only does Nageswaran's approach fail to teach such recited claim limitations, Nageswaran actually teaches away. Unlike Applicant's deques, Nageswaran instead teaches use of a dual pass process for reducing idle threads in a single queue 140 in which a first pass marks idle threads as "Being Removed" (Nageswaran, col.4:lines 6 - 7). Then, in a second pass, a certain number of the threads marked "Being Removed" are eliminated from the pool. (Nageswaran, col.4:lines 15 – 18). Nageswaran uses this "two phased solution" to enable their system to accommodate a sudden burst of traffic (Nageswaran, col. 4: lines 9 – 12: "It should be understood that threads 138 that are in the Being Removed state can still be reused, for example, if the thread manager 132 gets a spurt of requests that it has to service and that the thread has not yet been released.") (See also, col. 4:lines 21 – 25: "In between the transition from phase one to phase two, if the server thread manager senses a spurt where the number of request [sic] is larger than the X + (Y-X)/2, then the thread manager 132 can choose to abort the phase 2 commencement.") While the Office Action is correct that such a two-phased process could indeed be used in conjunction with Applicant's claimed embodiments, because such an addition is neither taught nor suggested by Nageswaran, the Office Action's inference to do so would require an impermissible exercise in hindsight. In conclusion, Nageswaran teaches away from the recited claim limitations as to its purpose as well as its approach.

Because Nageswaran's thread pool does holds execution threads, not connections, Nageswaran would not have considered adding a separate deque to manage resources that have been determined to be at least one of not created successfully and not able to be refreshed. Nageswaran's execution threads, in contrast to the recited computing resources, are available if in existence, and thus may be managed by Nageswaran's conventional uniform thread pool approach. Applicant respectfully suggests that since the inference to extend Nageswaran's approach to include managing resources that have been determined to be at least one of not created successfully and not able to be refreshed separately from resources that are available cannot be drawn from Nageswaran's teachings, the Office Action must have drawn such inference from the present application through an impermissible use of hindsight.

Further, since Nageswaran is relying on execution threads in the resource pool as always being either "in use" or "idle" (Nageswaran, col.4:lines 4 – 5: All threads 138 currently marked as in use are obviously not eligible to be deleted), any argued addition of the recited mechanism for managing resources determined to be at least one of not created successfully and not able to be refreshed separately from available resources would render Nageswaran inoperable or change Nageswaran's principle of operation (see MPEP § 2143.01).

In sum, Nageswaran fails to teach, suggest or otherwise render obvious the embodiments recited by claim 1.

Claim 11

Nageswaran fails to render the embodiments of amended claim 11 unpatentable at least for failing to teach, suggest or otherwise render obvious "determining whether at least one of the resources is functioning properly" recited by amended claim 11:

> 11. (Currently Amended) A method for performing resource pool maintenance for an application server, comprising:
>
> triggering a test for pool resources;
>
> determining whether at least one of the resources is functioning properly by performing a test on pool resources; and

refreshing pool resources based on the pool resources testing.

As noted in Applicant's specification, "Resources are tested at step 320. In one embodiment, the resource test determines if the resource is functioning properly," (Specification, paragraph [0024], emphasis added). Nageswaran's approach not only fails to disclose such claim limitation, Nageswaran actually teaches away by disclosing instead the simple removal of execution threads from the resource pool if the threads become idle, "[t]hreads 138 that are not dedicated for any particular transaction are prime candidates to be released and thread manager 132 would identify these threads and mark their state as "Being Removed") (Nageswaran, col. 3:lines 54 – 67 and col. 4: lines 1 – 18).

Further, any argued addition of the recited mechanism for testing resources to Nageswaran's approach would render Nageswaran inoperable or change Nageswaran's principle of operation (see MPEP § 2143.01) because Nageswaran presently REMOVES idle threads from the thread pool – nothing of the thread is left to be refreshed.

Further, extending Nageswaran to include refreshing threads requires an impermissible exercise in hindsight to incorporate such an inference from the present application into Nageswaran.

Claim 14

Amended claim 14 also recites limitations that are neither taught, suggested or otherwise rendered obvious by Nageswaran, as argued above with respect to amended claim 1 and is therefore allowable over Nageswaran for at least the same reasons:

14. (Currently Amended) A method for performing maintenance on connection pool deques in an application server, comprising:

traversing an unavailable deque of resources that have been determined to be at least one of not created successfully and not able to be refreshed and a reserved deque of resources that are allocated;

performing maintenance on the unavailable deque; and

performing maintenance on the reserved deque.

Rejections under 35 U.S.C. § 103(a)

Claims 2 – 3 and 12

In Item 13, pages 6-7, Claims 2-3 and 12 are rejected under 35 U.S.C 103(a) as being unpatentable over Nageswaran, in view of June et al. (US 2004/0045008A1), herein after referred as June. Applicant respectfully traverses.

Nageswaran's failures to teach, suggest or otherwise render obvious the embodiments recited by claims 1 and 11 were discussed previously. Since claims 2 – 3 and 12 depend from claims 1 and 11 respectfully, either directly or indirectly, Nageswaran cannot render the embodiments of claims 2 – 3 and 12 unpatentable if Nageswaran fails to teach, suggest or otherwise render obvious the embodiments recited by claims 1 and 11.

Since Nageswaran fails to make any suggestion addressing resources that have been determined to be at least one of not created successfully and not able to be refreshed, the Office Action's inference to combine June's connector architecture with Nageswaran must have been drawn from the present application through an impermissible exercise in hindsight.

Additionally, even if the Office Action were even allowed to make such a combination, such asserted combination still would not teach separate management of connections using available and unavailable deques.

Since claims 2 – 3 and 12 depend from claims 1 and 11 respectfully, either directly or indirectly, the asserted combination of June with Nageswaran, even if possible, could not render the embodiments of claims 2 – 3 and 12 unpatentable because the combination of June and Nageswaran does not render the embodiments recited by claims 1 and 11 unpatentable.

Claims 16 – 19

In Item 16, page 7 – 9, Claims 16 - 19 are rejected under 35 U.SC. 103(a) as being unpatentable over Nageswaran, in view of Sharma et al. (US 6,182,109) herein after referred as

Sharma. Applicant respectfully traverses.

Since claims 16 – 17 depend from claim 14, either directly or indirectly, Nageswaran cannot render the embodiments of claims 16 – 17 unpatentable if Nageswaran fails to teach, suggest or otherwise render obvious the embodiments recited by claim 14. Nageswaran's failure to teach, suggest or otherwise render obvious the embodiments recited by claim 14 was discussed previously.

Applicants respectfully submit that the Office Action's asserted combination of Nageswaran with Sharma, even if such a combination were even possible, fails to remedy the shortcomings of Nageswaran in failing to teach, suggest or otherwise render obvious the embodiments recited by amended claim 14 at least regarding the recited, "traversing an unavailable deque of resources that have been determined to be at least one of not created successfully and not able to be refreshed and a reserved deque of resources that are allocated." Sharma's execution thread manager instead populates a single pool of execution threads. (Sharma, Abstract).

Further, Sharma appears to address an entirely different problem. Sharma states their system reduces latency arising during switching between execution threads in a server [col. 21:lines 30 - 40], which is an entirely different problem than that addressed by the recited "unavailable deque of resources that have been determined to be at least one of not created successfully and not able to be refreshed and a reserved deque of resources that are allocated" recited in the preamble of claim 14. Accordingly, Sharma teaches away as to its purpose, as well as its operation.

Even if Sharma, arguendo, did address resources that have been determined to be at least one of not created successfully and not able to be refreshed, since Nageswaran fails to make any suggestion addressing such problems, the Office Action's inference to combine Sharma with Nageswaran must have been drawn from the present application through an impermissible exercise in hindsight.

Since claims 16 - 17 depend from claim 14, either directly or indirectly, the asserted combination of Sharma with Nageswaran, even if possible, could not render the embodiments of

claims 16 - 17 unpatentable because the combination of Sharma and Nageswaran does not render the embodiments recited by claim 14 unpatentable.

Claims 18 - 19

Amended claim 18 is patentable over the asserted combination of Nageswaran and Sharma at least for reciting either "determining that the resource was created successfully" and "moving the resource to an available deque; when successful generation of the resource is confirmed, otherwise moving the resource to an unavailable deque for tracking resources that are not created successfully," neither of which is taught, suggested or otherwise rendered obvious by that asserted combination:

> 18. (Currently Amended) A method for performing resource creation in a connection pool in an application server, comprising:
>
> generating a resource in connection pool;
>
> determining that the resource was created successfully in the connection pool; and
>
> moving the resource to an available deque; when successful generation of the resource is confirmed, otherwise moving the resource to an unavailable deque for tracking resources that are not created successfully.

Like Nageswaran, Sharma also fails to consider the problem of determining whether an attempt at generating a resource is successfully completed and tracking resources that have not been created successfully. Accordingly, the asserted combination with Nageswaran, if even possible, fails to teach suggest or otherwise render obvious the embodiments encompassed by amended claim 18. Therefore, for at least these reasons, the Applicant respectfully submits that claim 18 is allowable over the art of record and is in condition for allowance.

Since claim 19 depends from claim 18, the asserted combination of Nageswaran and Sharma cannot render the embodiments of claim 19 unpatentable if the asserted combination fails to render the embodiments recited by claim 18 unpatentable.

Claim 20

Amended claim 20 is allowable for at least the reasons argued above with respect to claims 1, 11, 14 and 18.

Claims 4 - 10, 13 and 15

Claims 4 - 10, 13 and 15 are dependent upon claims 1, 11 and 14 respectively, and thus include each and every feature of the corresponding independent claims. Each of claims 4 - 10, 13 and 15 is therefore allowable for the reasons given above for the claims 1, 11 and 14. In addition, each of claims 4 - 10, 13 and 15 introduces one or more additional limitations that independently render it patentable. Therefore, it is respectfully submitted that claims 4 - 10, 13 and 15 are allowable for the reasons given above with respect to claims 1, 11 and 14.

Conclusion

Because Nageswaran, alone or in any combination with either or both of June and Sharma, fails to consider recited claim limitations; would be rendered inoperable for its intended purpose or require changes to its principles of operation; require impermissible hindsight; and teach away from the embodiments recited by claims 1 - 20 as to their structure as well as its purpose, Applicant's respectfully request: 1) removal of the rejections; 2) removal of Nageswaran from further consideration as a reference in the present case; and 3) continuation of prosecution or allowance of the claims.

The references cited by the Examiner but not relied upon have been reviewed, but are not believed to render the claims unpatentable, either singly or in combination.

The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: 7/20/2005

By: _____

Paul A. Durdik
Reg. No. 37,819

FLIESLER MEYER LLP
Four Embarcadero Center, Fourth Floor
San Francisco, California 94111-4156
Telephone: (415) 362-3800 x227
Facsimile: (415) 362-2928